Alphabets and related topics

We will describe in this section the fundamental building blocks of languages. Starting with *alphabets*.

Reading

Sections 0.2 (pages 13 and on), 0.3, 0.4

Alphabets

An **alphabet** is any *nonempty* finite set. It is essentially the set of "characters" that we will deal with. It may be as simple as the two elements "0" and "1", or it could be as complex as the different keywords in a programming language.

As an example, consider what happens when the language's compiler is reading a new file:

- First the "lexer" runs, whose job is to collect together the characters that form "words", and identify them as keywords, identifiers, numbers, operator symbols etc. As far as the lexer is concerned, its "alphabet" is all valid characters (spaces, tabs, newlines, alphanumeric characters, most punctuation).
- The output of the lexer is then passed over to the "parser", whose job it is to form meaningful language expressions out of the outputs of the lexer, which are usually called tokens. The "alphabet" for the parser would consist of elements like "FOR", "ELSE", "ID", "INTEGER", "STRING", "LET", "IN", "(" and so on.

A **string** in an alphabet is any list of symbols from the alphabet, possibly empty. The order of the symbols matters. Some times strings are also called "words", and letters like w and v are used to denote them. The **length** of the string is the number of symbols in it. The **empty string** is typically denoted by the greek letter epsilon: `\epsilon`.

Given two strings, we can ask if one is a *substring* of the other. This means that the string must appear consecutively somewhere within the other string.

Question: How many substrings does the string "abc" have? List them.

A common operation for strings is **concatenation**, where the two strings are placed next to each other, the second one starting where there first left off.

Two other less frequent terms are *prefix* and *suffix*. A string w is a **prefix** of another string v, if v is the result of concatenating w with some other string. w is a **suffix** of v, if v is the result of concatenating some other string with w.

Question: What are all the prefixes of "abc"? What are all its suffixes?

A **language** is simply any set of strings. It is essentially used to represent those strings that are "meaningful" for the problem at hand. For example the language for the lexer we described earlier would consist of all the "words" that are valid in the language.

We will always assume that the symbols in an alphabet have a natural **order** to them. Based on that order we can then define a **lexicographic ordering** on strings:

> A string w is less than a string v if: - w is a prefix string of v, or - At the first place where they disagree, the symbol in w is less than the corresponding symbol in v